Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	112	frequency near5 (cache adj miss$2)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:39
S2	25	frequency near5 (cache adj miss$2) and (optimis$3 or optimiz$3 or improv$4) and (prefetch$3 or pre-fetch$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:39
S3	196	717/153	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:39
S4	0	frequency near5 (cache adj miss$2) and 15?/???	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:40
S5	10	(cache adj miss$2) and 15?/???	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:41
S6	2287	(cache adj miss$2) and (pre-fetch$3 or prefetch$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:41
S7	423	((cache adj miss$2) and (pre-fetch$3 or prefetch$3)) same performance	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:43
S8	154	((cache adj miss$2) and (pre-fetch$3 or prefetch$3)) same performance and degrad$5	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:44
S9	12	((cache adj miss$2) and (pre-fetch$3 or prefetch$3)) same performance and (S2 or S3 or I2/I3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:43

S10	107	((cache adj miss$2) and (pre-fetch$3 or prefetch$3)) same performance and degrad$5 and (optimiz$5 or optimis$5)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:50
S11	6	S9 and S10	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:50
S12	1	((cache adj miss$2) and (pre-fetch$3 or prefetch$3)) same performance and degrad$5 and (optimiz$5 or optimis$5) and (cyclic$4 near5 graph)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:50
S13	220	cyclic$4 near5 graph and (optimis$5 or optimiz$5)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:51
S14	2	(cyclic$4 near5 graph) and (optimis$5 or optimiz$5) and (prefetch$3 or pre-fetch$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:52
S15	4	(cyclic$4 near5 dependen$2) and (optimis$5 or optimiz$5) and (prefetch$3 or pre-fetch$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 11:59
S16	5	(stride near5 dependen$2) and (optimis$5 or optimiz$5) and (prefetch$3 or pre-fetch$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 12:01
S17	10	("5797013" "5704053" "6539541" "5862385" "5950007").pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/11/30 12:02

1 **Reference distance as a metric f r data l cality**

Changwoo Pyo; Kyung-Woo Lee; Hye-Kyung Han; Gyungho Lee;

High Performance Computing on the Information Superhighway, 1997. HPC Asia '97 , 28 April-2 May 1997
Pages:151 - 156

[Abstract] [PDF Full-Text (380 KB)] IEEE CNF

8 **On predictability and ptimizati n of multipr grammed caches f r real-time applicati ns**

Shahrier, S.M.; Juh-Charn Liu;

Performance, Computing, and Communications Conference, 1997. IPCCC 1997., IEEE International , 5-7 Feb. 1997
Pages:17 - 25

[Abstract] [PDF Full-Text (796 KB)] IEEE CNF

Terms used **optimize cache prefetch** Found **10,248** of **148,162**

Sort results by relevance Save results to a Binder Try an Advanced Search

Display results expanded form Search Tips Try this search in The ACM Guide

☐ Open results in a new window

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 The Performance of Runtime Data Cache Prefetching in a Dynamic Optimization System

Jiwei Lu, Howard Chen, Rao Fu, Wei-Chung Hsu, Bobbie Othmer, Pen-Chung Yew, Dong-Yuan Chen

December 2003 **Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture**

Full text available: pdf(253.79 KB) Publisher Site Additional Information: full citation, abstract, citings, index terms

Traditional software controlled data cache prefetching isoften ineffective due to the lack of runtime cache miss andmiss address information. To overcome this limitation, weimplement runtime data cache prefetching in the dynamicoptimization system ADORE (ADaptive Object code RE-optimization).Its performance has been compared withstatic software prefetching on the SPEC2000 benchmarksuite. Runtime cache prefetching shows better performance.On an Itanium 2 based Linux workstation, it can increasepe ...

2 Research sessions: implementation techniques: Fractal prefetching B+-Trees: optimizing both cache and disk performance

Shimin Chen, Phillip B. Gibbons, Todd C. Mowry, Gary Valentin

June 2002 **Proceedings of the 2002 ACM SIGMOD international conference on Management of data**

Full text available: pdf(1.49 MB) Additional Information: full citation, abstract, references, citings, index terms

B+-Trees have been traditionally optimized for I/O performance with disk pages as tree nodes. Recently, researchers have proposed new types of B+-Trees optimized for CPU cache performance in main memory environments, where the tree node sizes are one or a few cache lines. Unfortunately, due primarily to this large discrepancy in optimal node sizes, existing disk-optimized B+-Trees suffer from poor cache performance while cache-optimized B+-Trees exhibi ...

3 Instruction prefetching of systems codes with layout optimized for reduced cache misses

Chun Xia, Josep Torrellas

May 1996 **ACM SIGARCH Computer Architecture News , Proceedings f the 23rd annual internati nal symp sium n C mputer architecture,** Volume 24 Issue 2

Full text available: pdf(1.65 MB) Additional Information: full citation, abstract, references, citings, index terms

High-performing on-chip instruction caches are crucial to keep fast processors busy.

Unfortunately, while on-chip caches are usually successful at intercepting instruction fetches in loop-intensive engineering codes, they are less able to do so in large systems codes. To improve the performance of the latter codes, the compiler can be used to lay out the code in memory for reduced cache conflicts. Interestingly, such an operation leaves the code in a state that can be exploited by a new type of ...

4 Evaluating the impact of memory system performance on software prefetching and locality optimizations

Abdel-Hameed A. Badawy, Aneesh Aggarwal, Donald Yeung, Chau-Wen Tseng

June 2001 **Proceedings of the 15th international conference on Supercomputing**

Full text available: pdf(387.06 KB) Additional Information: full citation, abstract, references, citings, index terms

Software prefetching and locality optimizations are techniques for overcoming the speed gap between processor and memory. In this paper, we evaluate the impact of memory trends on the effectiveness of software prefetching and locality optimizations for three types of applications: regular scientific codes, irregular scientific codes, and pointer-chasing codes. We find for many applications, software prefetching outperforms locality optimizations when there is sufficient memory bandwidth, but ...

5 Context-based prefetch – an optimization for implementing objects on relations

Philip A. Bernstein, Shankar Pal, David Shutt

December 2000 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 9 Issue 3

Full text available: pdf(142.24 KB) Additional Information: full citation, abstract, index terms

When implementing persistent objects on a relational database, a major performance issue is prefetching data to minimize the number of round-trips to the database. This is especially hard with navigational applications, since future accesses are unpredictable. We propose the use of the context in which an object is loaded as a predictor of future accesses, where a context can be a stored collection of relationships, a query result, or a complex object. When an object O's state is loaded, similar ...

Keywords: Caching, Object-oriented database, Object-relational mapping, Prefetch